Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs

Generate OACS

Search Results - Record(s) 1 through 2 of 2 returned.

□ 1. Document ID: US 6741959 B1

Using default format because multiple data bases are involved.

```
  L15: Entry 1 of 2                        File: USPT                    May 25, 2004

US-PAT-NO: 6741959
DOCUMENT-IDENTIFIER: US 6741959 B1
** See image for Certificate of Correction **

TITLE: System and method to retrieving information with natural language queries

DATE-ISSUED: May 25, 2004

INVENTOR-INFORMATION:
```

NAME	CITY	STATE	ZIP CODE	COUNTRY
Kaiser; Matthias	Mountain View	CA		

```
US-CL-CURRENT: 704/7; 704/9, 707/3, 707/4, 707/5
```

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

□ 2. Document ID: US 6574632 B2

```
  L15: Entry 2 of 2                        File: USPT                    Jun 3, 2003

US-PAT-NO: 6574632
DOCUMENT-IDENTIFIER: US 6574632 B2
** See image for Certificate of Correction **

TITLE: Multiple engine information retrieval and visualization system

DATE-ISSUED: June 3, 2003

INVENTOR-INFORMATION:
```

NAME	CITY	STATE	ZIP CODE	COUNTRY
Fox; Kevin L.	Palm Bay	FL		
Frieder; Ophir	Chicago	IL		
Knepper; Margaret M.	Indialantic	FL		
Killam; Robert A.	Melbourne	FL		
Nemethy; Joseph M.	Melbourne	FL		
Cusick; Gregory J.	Melbourne Beach	FL		
Snowberg; Eric J.	Castle Rock	CO		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Harris Corporation	Melbourne	FL			02

APPL-NO: 09/ 195773 [PALM]
DATE FILED: November 18, 1998

INT-CL: [07] G06 F 17/30, G06 F 15/16

US-CL-ISSUED: 707/102; 707/5, 707/10, 709/218
US-CL-CURRENT: 707/102; 707/10, 707/5, 709/218

FIELD-OF-SEARCH: 707/2, 707/3-6, 707/10, 707/100, 707/102, 707/513, 709/217-219, 709/203, 705/35-38, 345/764

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
4599692	July 1986	Tan et al.	364/513
5062143	October 1991	Schmitt	382/36
5325298	June 1994	Gallant	364/419.19
5649193	July 1997	Sumita et al.	395/614
5675819	October 1997	Schuetze	395/760
5696962	December 1997	Kupiec	395/604
5706497	January 1998	Takahashi et al.	395/605
5713016	January 1998	Hill	395/605
5717913	February 1998	Driscoll	395/605
5724567	March 1998	Rose et al.	395/602
5724571	March 1998	Woods	395/605
5745893	April 1998	Hill et al.	707/5
5765150	June 1998	Burrows	707/5
5774888	June 1998	Light	707/1
5835905	November 1998	Pirolli et al.	707/102
5987446	November 1999	Corey et al.	707/3
6026397	February 2000	Sheppard	707/2
6029172	February 2000	Jorna et al.	707/102
6038561	March 2000	Snyder et al.	707/10
6041331	March 2000	Weiner et al.	707/103R
6216134	April 2001	Heckerman et al.	704/202
6269362	July 2001	Broder et al.	707/1

OTHER PUBLICATIONS

"Displaying Data in Multidimensional Relevance Space with 2D Visualization Maps," Assa et al., Proceedings of the 1997 Visualization Conference, Phoenix, Arizona, USA, ACM Press, pp. 127-134.*
Shaw et al., Combination of Multiple Searches, Department of Computer Science, Virginia Tech., Blacksburg, Virginia, 4 pages Date Unknown.

Cavnar, Using an N-Gram-Based Document Representation with a Vector Processing Retrieval Model, Ann Arbor, Michigan, 9 pages Date Unknown.

ART-UNIT: 2172

PRIMARY-EXAMINER: Alam; Hosain T.

ATTY-AGENT-FIRM: Allen, Dyer, Doppelt, Milbrath & Gilchrist, P.A.

ABSTRACT:

An information retrieval and visualization system utilizes multiple search engines for retrieving documents from a document database based upon user input queries. Search engines include an n-gram search engine and a vector space model search engine using a neural network training algorithm. Each search engine produces a common mathematical representation of each retrieved document. The retrieved documents are then combined and ranked. Mathematical representations for each respective document is mapped onto a display. Information displayed includes a three-dimensional display of keywords from the user input query. The three-dimensional visualization capability based upon the mathematical representation of information within the information retrieval and visualization system provides users with an intuitive understanding, with relevance feedback/query refinement techniques that can be better utilized, resulting in higher retrieval accuracy (precision).

27 Claims, 22 Drawing figures

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw. D

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Term	Documents
GENERIC	203487
GENERICS	546
(13 AND GENERIC).PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD.	2
(L13 AND GENERIC).PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD.	2

Display Format: - Change Format

Previous Page Next Page Go to Doc#

Freeform Search

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term: L13 and generic

Display: 50 **Documents in Display Format:** FRO **Starting with Number** 1

Generate: ○ **Hit List** ● **Hit Count** ○ **Side by Side** ○ **Image**

Search Clear Interrupt

Search History

DATE: Friday, September 16, 2005 Printable Copy Create Case

Set Name **side by side**	**Query**	**Hit Count**	**Set Name** **result set**
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L15	L13 and generic	2	L15
L14	L13 and occurence	1	L14
L13	L12 and frequency	28	L13
L12	l8 and (word near group$)	40	L12
L11	l8 and "non-generic word"	0	L11
DB=USPT; PLUR=YES; OP=OR			
L10	L9 and proximate$	0	L10
L9	L8 and (generic near word$)	3	L9
L8	L7 and (natural near language)	276	L8
L7	707/5.ccls.	1418	L7
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L6	L5 and (xml near storage)	6	L6
L5	xml near exchange	191	L5
L4	L3 and import$	6	L4
L3	L1 and (data near aggregat$)	9	L3
L2	L1 and (data near aggregator)	2	L2

L1	xml near storage	150	L1

END OF SEARCH HISTORY

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

End of Result Set

□ Generate Collection Print

L14: Entry 1 of 1 File: USPT Jun 28, 1994

DOCUMENT-IDENTIFIER: US 5325298 A
TITLE: Methods for generating or revising context vectors for a plurality of word stems

Abstract Text (1):

A method for generating context vectors for use in a document storage and retrieval system. A context vector is a fixed length list of component values generated to approximate conceptual relationships. A context vector is generated for each word stem. The component values may be manually determined on the basis of conceptual relationships to word-based features for a core group of word stems The core group of context vectors are used to generate the remaining context vectors based on the proximity of a word stem to words and the context vectors assigned to those words. The core group may also be generated by initially assigning each core word stem a row vector from an identity matrix and then performing the proximity based algorithm. Context vectors may be revised as new records are added to the system, based on the proximity relationships between word stems in the new records.

Brief Summary Text (5):

Waltz and Pollack, in their article entitled "Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation" in Cognitive Science, Vol. 9, pages 51-74 (1985), presented a neural network based model for word sense disambiguation using high level features which are associated with "micro-features". The system was implemented by running several iterations of spreading activation which would be computationally inefficient for medium-or large-scale systems.

Detailed Description Text (11):

It may be possible to enhance the accuracy of the searching techniques by using additional processing on the records. For example, a parsing algorithm can be used to identify the subject, predicate and verb in each sentence. The subject and verb or the subject, verb and predicate can then be assigned 38 a greater weight than the other words in each sentence. Another method is to give the first 100 (or so) words in a record extra weight. Other methods of assigning weights 38 to words in a record may also be used. There are well known algorithms based on the frequency of use of a word in a record or in a series of records which may be used so as to assign a different weight to each of the remaining words in the record. For example, (1, p. 304) stem s in record d might be weighted by

Detailed Description Text (41):

A core group of word stems are taken from the list. In accordance with a preferred method, word stems having the highest information content are taken from the top of the list. For example, the first 1,000 word stems having the highest information content may be selected. For the core group of word stems however selected, context vectors may be generated by hand 108. Temporarily a zero (0) vector is assigned to any other word stems remaining 110. A word stem w which has temporarily been assigned a zero (0) vector is then taken. The word stem with the highest information content is selected in a preferred method. For this word stem, the context vectors of word stems that are close to w in the training corpus records are weighted by their distance from w. For example, the 10 stems preceding and following each occurrence of the word stem may be used. The weighted context vectors are added up to produce a context vector for the word stem 112. The context vector can then be normalized 114. The resulting context vector becomes w's permanent context vector. The next word stem w from those word stems which have only a temporary 0 vector is then selected and the process is repeated 116. It is recommended that at least 1000 records be used. Once the dictionary of context vectors is completed, the invention may be used to its full benefit. For such automatic dictionary building, multiple meanings of a word do not enter in; all stems have only one context vector.

Detailed Description Text (44):
In accordance with the presently preferred weighting method, nine (9) word stems are taken from each side of the word stem w. If any of these word stems are identical to word stem w they can be left out of the computation. Indeed, only the core group word stems with assigned row vectors are used in the computation. When two occurences of word stem w are close to one another, the nine word stems on either side may overlap. In this case, their row vectors will be weighted and summed twice. The preferred weighting scheme is to multiply a word row vector by .eta. (distance/3), where .eta. (x) is the probability density function for the normal distribution at a distance of x standard deviations from the mean. The normal distribution is the well-known Gaussian bell curve. Every three word stems away is treated as a standard deviation. Thus, for example, the row vector for a word that is six word stems away from word stem w is multiplied by the value of the normal distribution at two standard deviations. The weighted row vectors are summed to obtain a sum vector.

Detailed Description Text (45):
The sum vector just obtained is normalized 126 so that all vectors being formed will have the same total weight. The normalized vector is assigned as the context vector for the word stem w 128. Forming the weighted sum of row vectors is repeated 130 for each word stem using the original row vectors to compute the context vectors. Even if a word stem has been assigned a context vector, its original row vector is used in the computation. Alternatively, it would be possible to use the context vectors as they are assigned, however, in this case it may be advisable to go through the process at least a second time for each core group word stem so that all of the core context vectors may be based on vectors generated by the system rather than having some predominantly based on the original row vectors and others based more on the generated context vectors.

Detailed Description Text (46):
After all of the word stems in the core group have been assigned context vectors, the core group of context vectors may be used in the dictionary building algorithm of FIG. 11. In this manner, a dictionary of context vectors is built using entirely automatic methods. Using this method of generating core group context vectors with an identity matrix, the original core group of word stems forms the features associated with each component value in the context vectors for the entire dictionary. A more random method of assigning the initial row vectors might be used but in that case, a feature may not be readily identified for each component value. In any case, the method of the present invention generates a context vector which is relationship based as it is grounded upon the proximity of words to one another in a training corpus of records.

Detailed Description Text (48):
A core group of context vectors is generated 306. This may be accomplished using manual entry of component values based on relationships between the word stem and the features or by the automatic method of FIG. 12. Each word stem in the core group is flagged 307 so that the core group word stems can be readily identified by the system. A zero vector is temporarily assigned 308 to all the remaining word stems found in the corpus of records. Each remaining word stem has a counter

associated with it for keeping track of the number of times that the word stem appears in the records in the system. As records are added to the system the counters are revised. If the system also revises context vectors for the core group word stems, these too would have counters associated therewith. The counters are initialized at zero 309.

Detailed Description Text (51):
After all word stems in the corpus of training records have been assigned context vectors, a dynamic revision of the context vectors can be made when an additional record is added to the system. It is presently preferred that the core group of word stems maintain their originally provided context vectors. However, a dynamic system of the present invention could be implemented in which the core group of word stems as well as any remaining word stems are all available for updating in the dynamic system.

Detailed Description Paragraph Table (3):

APPENDIX

(1) Salton, G. Automatic text processing: The transformation, analysis, and retrieval of information by computer. Reading Ma.: Addison-Wesley, 1989. (2) Rumelhart, D. E. & McClelland, J. L. (Eds.) Parallel Distributed Processing: Explorations in the Microstructures of Congntion, Vol. 1 and Vol. 2 MIT Press 1986. (3) Anderson, J. A. and Rosenfeld, E. (eds). Neurocomputing, A reader, MIT Press, 1988. (4) Hecht-Nielson, Neurocomputing. Reading Ma.: Addison-Wesley, 1990. (5) Waltz, D. L. & Pollack, J., Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation. Cognitive Science 9, 51-74 (1985). (6) Cottrell, G. W. Connectionist Parsing, Seventh Annual Conference of the Cognitive Science Society, Irvine, CA. (7) Belew, R. K., Adaptive Information Retrieval. 12th International Conference on Research & Development in Information Retrieval. Boston, June, 1989. (8) Samet, H. The Design and Analysis of Spatial Data Structures. Reading, Ma: Addison-Wesley Publishing Company, 1990. (9) Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., Harshman,, R. Indexing by Latent Semantic Analysis, Journal of the American Society for Info. Science, 41(b):391-407, 1990. (10) Bein, J. & Smolensky, P. (1988). Application of the interactive activation model to document retrieval. Proceedings of Neuro-Nimes 1988: Neural networds and their applications. Nimes, France, November 395-308. (11) MacQueen, J. B. Some Methods for Classification and Analysis of Multivariate Observations. Proc. Symp. Math. Statist. and Probability, 5th, Berkeley, 1, 281-297, AD 669871, University of California Press, Berkeley.

Current US Cross Reference Classification (1):
707/5

Other Reference Publication (1):
Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation, Waltz, Polack.

CLAIMS:

1. A method for generating a dictionary of context vectors comprising:

providing a corpus of records, each including a series of words wherein each word corresponds to one of a plurality of word stems;

generating a context vector for each of a core group of word stems;

temporarily assigning a zero context vector to the remaining word stems in said plurality of word stems that are not in said core group;

for each word stem with a zero vector, combining context vectors based on proximity in each of said series of words between the word corresponding to said word stem and the words corresponding to said context vectors to generate a context vector for said word stem.

4. The method of claim 1 wherein said step of generating context vectors for each word stem of a core group of word stems comprises selecting a series of features and assigning component values based on conceptual relationships between said word stem and said features.

5. The method of claim 1 wherein said step of generating context vectors for each word stem of a core group of word stems comprises assigning a different initial vector to each of said word stems, assigning weights to the context vectors corresponding to words appearing in a record with the word corresponding to said each word stem based on relative proximity, multiplying the context vectors of said words by said weights to form weighted context vectors, summing said weighted context vectors, normalizing the sum of said weighted context vectors and assigning the normalized context vector as the context vector for said each word stem.

7. The method of claim 1 wherein said step of generating context vectors for each word stem of a core group of word stems comprises assigning a different initial vector to each of said word stems, summing the context vectors corresponding to words appearing within a predetermined number of words from words corresponding to said each word stem in said corpus of records, normalizing the sum of said context vectors and assigning the normalized context vector as the context vector for said each word stem.

11. A method for generating a core group of context vectors comprising

providing a corpus of records, each including a series of words wherein each word corresponds to one of a plurality of word stems;

selecting a core group of word stems from said plurality of word stems;

assigning a different vector to each word stem in said core group of word stems; for each word stem in said core group, combining the different vectors based on proximity in each of said series of words between the word corresponding to said each word stem and the words corresponding to said different vectors to generate a context vector for said each word stem.

15. A method for generating a dictionary of context vectors comprising:

providing a corpus of records, each including a series of words wherein each word corresponds to one of a plurality of word stems;

generating a context vector for each of a core group of word stems;

temporarily assigning a zero context vector to the remaining word stems in said plurality of word stems that are not in said core group;

serially proceeding through the corpus of records and for each word stem that is not in said core group:

combining context vectors based on proximity in each of said series of words between the word corresponding to said word stem and the words corresponding to said context vectors to generate a sum vector for said word stem;

combining the sum vector with the context vector assigned to said word stem to generate a replacement context vector for said word stem.

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

```
L9: Entry 1 of 3                    File: USPT                    Jun 12, 2001
```

DOCUMENT-IDENTIFIER: US 6246977 B1
TITLE: Information retrieval utilizing semantic representation of text and based on constrained expansion of query words

Brief Summary Text (22):
Using a digital dictionary or thesaurus (also known as a "linguistic knowledge base") that identifies, for a particular sense of a word, senses of other words that are generic terms for the sense of the word ("hypernyms"), the invention changes the words within the logical forms produced by the parser to their hypernyms to create additional logical forms having an overall meaning that is hypernyms to the meaning of these original logical forms. For example, based on indications from the dictionary that a sense of "parent" is a hypernym of the ascribed sense of "father," a sense of "touch" is a hypernym of the ascribed sense of "hold," and a sense of "child" and sense of "person" are hypernyms of the ascribed sense of "baby," the invention might create additional logical forms as follows:

Detailed Description Text (3):
In a preferred embodiment, the conventional tokenizer shown in FIG. 1 is replaced with an improved information retrieval tokenization facility ("the facility") that parses input text to identify logical forms, then expands the logical forms using hypernyms. The invention overcomes the problems associated with conventional tokenization by parsing both indexed and query text to perform lexical, syntactic, and semantic analysis of this input text. This parsing process produces one or more logical forms, which identify words that perform primary roles in the query text and their intended senses, and that further identify the relationship between those words. The parser preferably produces logical forms that relate the deep subject, verb, and deep object of the input text. For example, for the input text "The father is holding the baby," the parser might produce logical form indicating the deep subject is "father," the verb is "hold," and the deep object is "baby." Because transforming input text into a logical form distills the input text to its fundamental meaning by eliminating modifiers and ignoring differences in tense and voice, transforming input text segments into the logical forms tends to unify the many different ways that may be used in a natural language to express the same idea. The parser further identifies the particular senses of these words in which they are used in the input text.

Detailed Description Text (4):
Using a digital dictionary or thesaurus (also known as a "linguistic knowledge base") that identifies, for a particular sense of a word, senses of other words that are generic terms for the sense of the word ("hypernyms"), the invention changes the words within the logical forms produced by the parser to their hypernyms to create additional logical forms having an overall meaning that is hypernymous to the meaning of these original logical forms. The invention then transforms all of the generated logical forms into tokens intelligible by the information retrieval system that compares the tokenized query to the index, and submits them to the information retrieval system.

Detailed Description Text (14):
As discussed above, because transforming input text into a logical form distills

the input text to its fundamental meaning by eliminating modifiers and ignoring differences in tense and voice, transforming input text segments into the logical forms tends to unify the many different ways that may be used in a natural language to express the same idea.

Current US Cross Reference Classification (3):
707/5

Other Reference Publication (3):
James Allen, "Natural Language Understanding," The Benjamin/Cummings Publishing Company, Inc., Chapter 8, pp. 227-238 (1995).

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L9: Entry 2 of 3 File: USPT Dec 12, 2000

DOCUMENT-IDENTIFIER: US 6161084 A

**** See image for Certificate of Correction ****

TITLE: Information retrieval utilizing semantic representation of text by identifying hypernyms and indexing multiple tokenized semantic structures to a same passage of text

Brief Summary Text (22):
Using a digital dictionary or thesaurus (also known as a "linguistic knowledge base") that identifies, for a particular sense of a word, senses of other words that are generic terms for the sense of the word ("hypernyms"), the invention changes the words within the logical forms produced by the parser to their hypernyms to create additional logical forms having an overall meaning that is hypernymous to the meaning of these original logical forms. For example, based on indications from the dictionary that a sense of "parent" is a hypernym of the ascribed sense of "father," a sense of "touch" is a hypernym of the ascribed sense of "hold," and a sense of "child" and sense of "person" are hypernyms of the ascribed sense of "baby," the invention might create additional logical forms as follows:

Detailed Description Text (3):
In a preferred embodiment, the conventional tokenizer shown in FIG. 1 is replaced with an improved information retrieval tokenization facility ("the facility") that parses input text to identify logical forms, then expands the logical forms using hypernyms. The invention overcomes the problems associated with conventional tokenization by parsing both indexed and query text to perform lexical, syntactic, and semantic analysis of this input text. This parsing process produces one or more logical forms, which identify words that perform primary roles in the query text and their intended senses, and that further identify the relationship between those words. The parser preferably produces logical forms that relate the deep subject, verb, and deep object of the input text. For example, for the input text "The father is holding the baby," the parser might produce logical form indicating the deep subject is "father," the verb is "hold," and the deep object is "baby." Because transforming input text into a logical form distills the input text to its fundamental meaning by eliminating modifiers and ignoring differences in tense and voice, transforming input text segments into the logical forms tends to unify the many different ways that may be used in a natural language to express the same idea. The parser further identifies the particular senses of these words in which they are used in the input text.

Detailed Description Text (4):
Using a digital dictionary or thesaurus (also known as a "linguistic knowledge base") that identifies, for a particular sense of a word, senses of other words that are generic terms for the sense of the word ("hypernyms"), the invention changes the words within the logical forms produced by the parser to their hypernyms to create additional logical forms having an overall meaning that is hypernymous to the meaning of these original logical forms. The invention then transforms all of the generated logical forms into tokens intelligible by the information retrieval system that compares the tokenized query to the index, and submits them to the information retrieval system.

Detailed Description Text (15):
As discussed above, because transforming input text into a logical form distills the input text to its fundamental meaning by eliminating modifiers and ignoring differences in tense and voice, transforming input text segments into the logical forms tends to unify the many different ways that may be used in a natural language to express the same idea.

Current US Cross Reference Classification (3):
707/5

Other Reference Publication (3):
James Allen, "Natural Language Understanding," The Benjamin/Cummings Publishing Company, Inc., Chapter 8, pp. 227-238 (1995).

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L9: Entry 2 of 3 File: USPT Dec 12, 2000

DOCUMENT-IDENTIFIER: US 6161084 A

**** See image for Certificate of Correction ****

TITLE: Information retrieval utilizing semantic representation of text by identifying hypernyms and indexing multiple tokenized semantic structures to a same passage of text

Brief Summary Text (22):

Using a digital dictionary or thesaurus (also known as a "linguistic knowledge base") that identifies, for a particular sense of a word, senses of other words that are generic terms for the sense of the word ("hypernyms"), the invention changes the words within the logical forms produced by the parser to their hypernyms to create additional logical forms having an overall meaning that is hypernymous to the meaning of these original logical forms. For example, based on indications from the dictionary that a sense of "parent" is a hypernym of the ascribed sense of "father," a sense of "touch" is a hypernym of the ascribed sense of "hold," and a sense of "child" and sense of "person" are hypernyms of the ascribed sense of "baby," the invention might create additional logical forms as follows:

Detailed Description Text (3):

In a preferred embodiment, the conventional tokenizer shown in FIG. 1 is replaced with an improved information retrieval tokenization facility ("the facility") that parses input text to identify logical forms, then expands the logical forms using hypernyms. The invention overcomes the problems associated with conventional tokenization by parsing both indexed and query text to perform lexical, syntactic, and semantic analysis of this input text. This parsing process produces one or more logical forms, which identify words that perform primary roles in the query text and their intended senses, and that further identify the relationship between those words. The parser preferably produces logical forms that relate the deep subject, verb, and deep object of the input text. For example, for the input text "The father is holding the baby," the parser might produce logical form indicating the deep subject is "father," the verb is "hold," and the deep object is "baby." Because transforming input text into a logical form distills the input text to its fundamental meaning by eliminating modifiers and ignoring differences in tense and voice, transforming input text segments into the logical forms tends to unify the many different ways that may be used in a natural language to express the same idea. The parser further identifies the particular senses of these words in which they are used in the input text.

Detailed Description Text (4):

Using a digital dictionary or thesaurus (also known as a "linguistic knowledge base") that identifies, for a particular sense of a word, senses of other words that are generic terms for the sense of the word ("hypernyms"), the invention changes the words within the logical forms produced by the parser to their hypernyms to create additional logical forms having an overall meaning that is hypernymous to the meaning of these original logical forms. The invention then transforms all of the generated logical forms into tokens intelligible by the information retrieval system that compares the tokenized query to the index, and submits them to the information retrieval system.

Detailed Description Text (15):
As discussed above, because transforming input text into a logical form distills the input text to its fundamental meaning by eliminating modifiers and ignoring differences in tense and voice, transforming input text segments into the logical forms tends to unify the many different ways that may be used in a natural language to express the same idea.

Current US Cross Reference Classification (3):
707/5

Other Reference Publication (3):
James Allen, "Natural Language Understanding," The Benjamin/Cummings Publishing Company, Inc., Chapter 8, pp. 227-238 (1995).

Previous Doc Next Doc Go to Doc#